Análisis de capacidad

Pruebas:

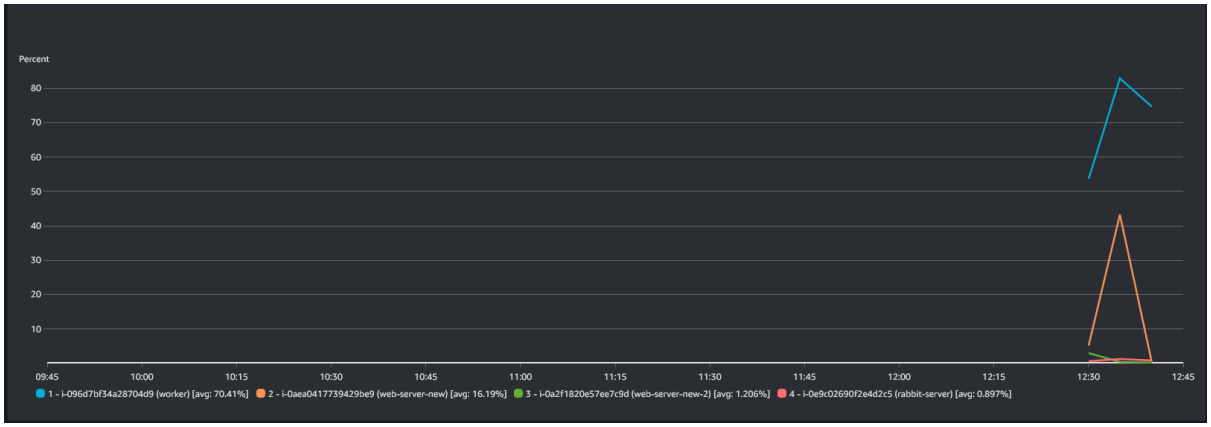
Las pruebas realizadas para esta iteración fueron tomadas y extendidas de la iteración anterior. En este caso utilizamos 8 hilos de ejecución cada uno realizando 20 pruebas en un tiempo distribuido entre 15 a 40 segundos. Para ello utilizamos el software para pruebas de rendimiento JMeter que hacía peticiones al proyecto alojado en la nube AWS.

Con el propósito de evidenciar los beneficios o efectos del auto escalamiento y el uso de balanceadores de carga, se hicieron 2 pruebas aisladas:

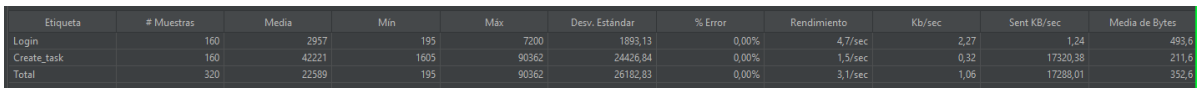
- Caso 1: Se utiliza una sola instancia y se hacían peticiones directamente.
- Caso 2: Se utilizan 2 instancias para simular auto escalamiento horizontal y se hace uso de un balanceador de cargas para distribuir las peticiones entre las dos instancias.

Caso 1:

Para este caso tenemos un comportamiento favorable donde el worker tiene un pico máximo de 80% de CPU mientras que la instancia que procesa las peticiones llega a un pico de 40% de uso de CPU:



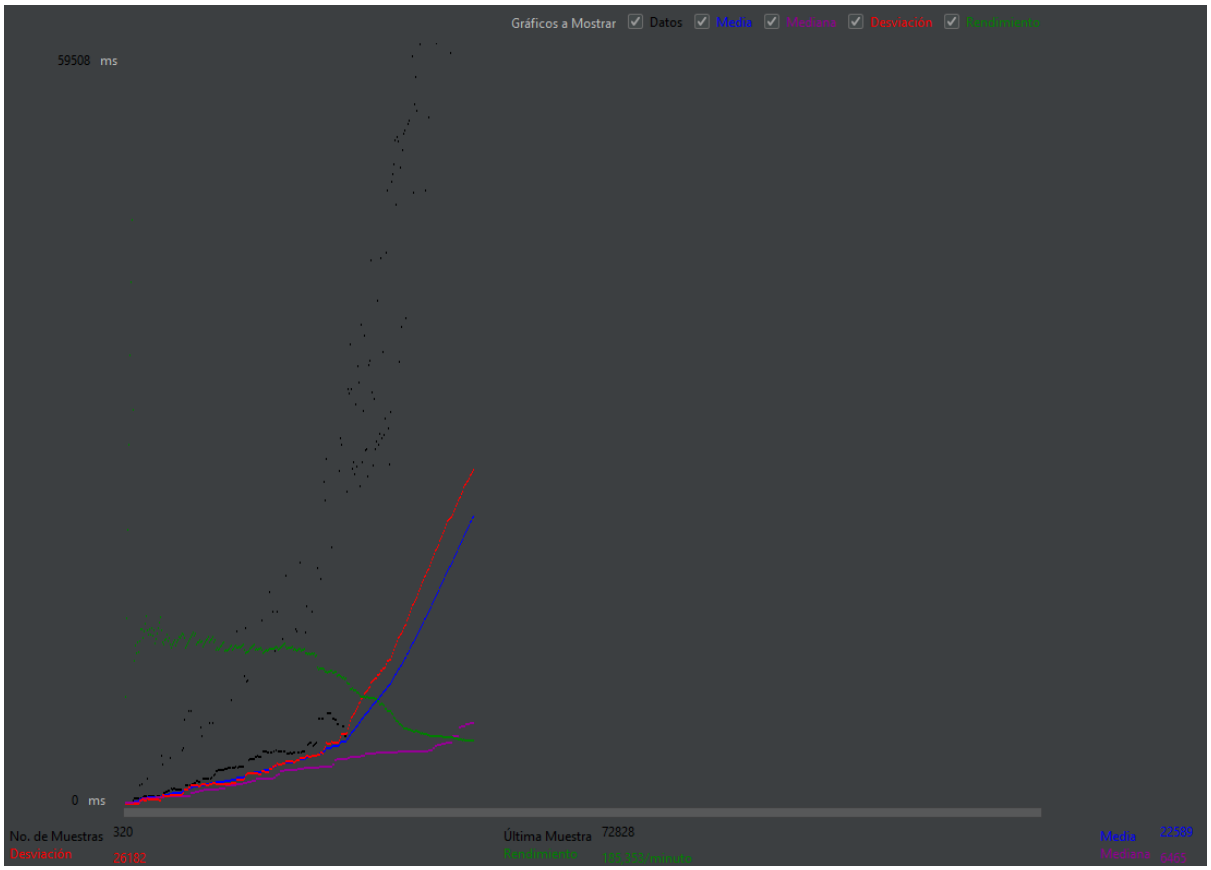
Entre las mediciones que genera JMeter se puede evidenciar un tiempo máximo de respuesta de 90 segundos. Esto refiriéndose a la API de crear una tarea de procesamiento de video. En general la media fueron 42 segundos lo cual no es ideal, es un buen rendimiento para el propósito de esta solución:



Etiqueta	# Muestras	Media	Min	Máx	Desv. Estándar	% Error	Rendimiento	Kb/sec	Sent KB/sec	Media de Bytes
Login	160	2957	195	7200	1893,13	0,00%	4,7/sec	2,27	1,24	493,6
Create_task	160	42221	1605	90362	24426,84	0,00%	1,5/sec	0,32	17320,38	211,6
Total	320	22589	195	90362	26182,83	0,00%	3,1/sec	1,06	17288,01	352,6

Etiqueta	# Muestras	Media	Min	Max	Desv. Estandar	% Error	Rendimiento (s)	Kb/sec	Sent KB/sec	Media Bytes
Login	160	2957	195	7200	1.893	0,00%	4,7	2,27	1,24	493,6
Create_task	160	42221	1605	90362	24.427	0,00%	1,5	0,32	17320,38	211,6
Total	320	22589	195	90362	26.183	0,00%	3,1	1,06	17288,01	352,6

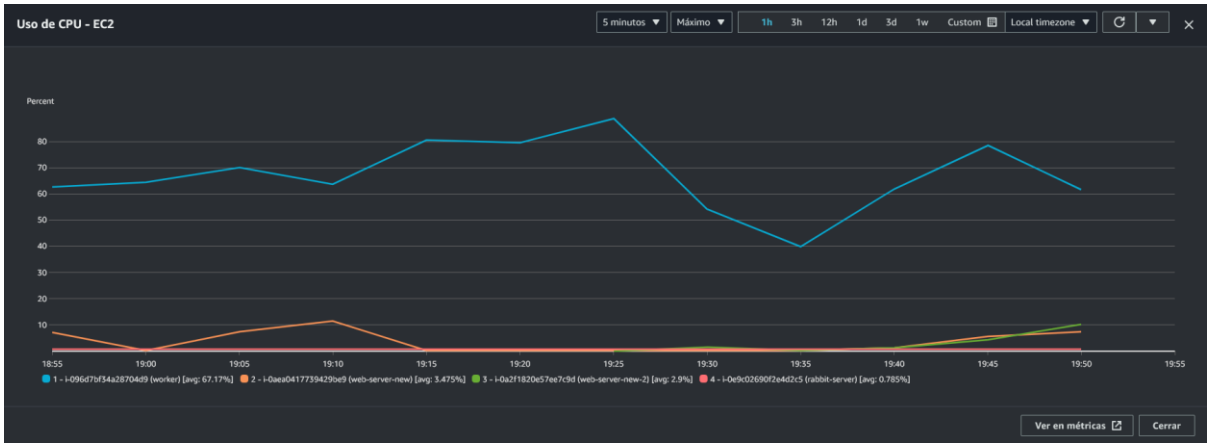
En la siguiente gráfica se ve de forma cómo se comporta el rendimiento y los tiempos de respuesta a lo largo de la prueba. En general, se entiende que el rendimiento no baja drásticamente como en las pruebas siguientes. Llegando a ser a lo largo del tiempo una solución y configuración más estable a futuro.



Rendimiento: 185,353/minuto

Caso 2:

Como se menciona anteriormente, por limitaciones del laboratorio, no es posible realizar un auto escalamiento directamente en la plataforma. Por tal razón, el escalamiento se simuló utilizando 2 instancias y un balanceador de cargas. La siguiente imagen es una gráfica tomada de AWS CloudWatch. Allí se puede evidenciar la prueba realizada en los últimos 10 minutos, de 19:40 a 19:50. Las instancias de color verde y naranja son las máquinas virtuales que reciben las peticiones y procesan la información de estas. Estas máquinas virtuales llegan a un consumo de CPU no mayor al 10% mientras que la línea azul representa al worker que procesa los videos tiene un pico de 80% de consumo de CPU:

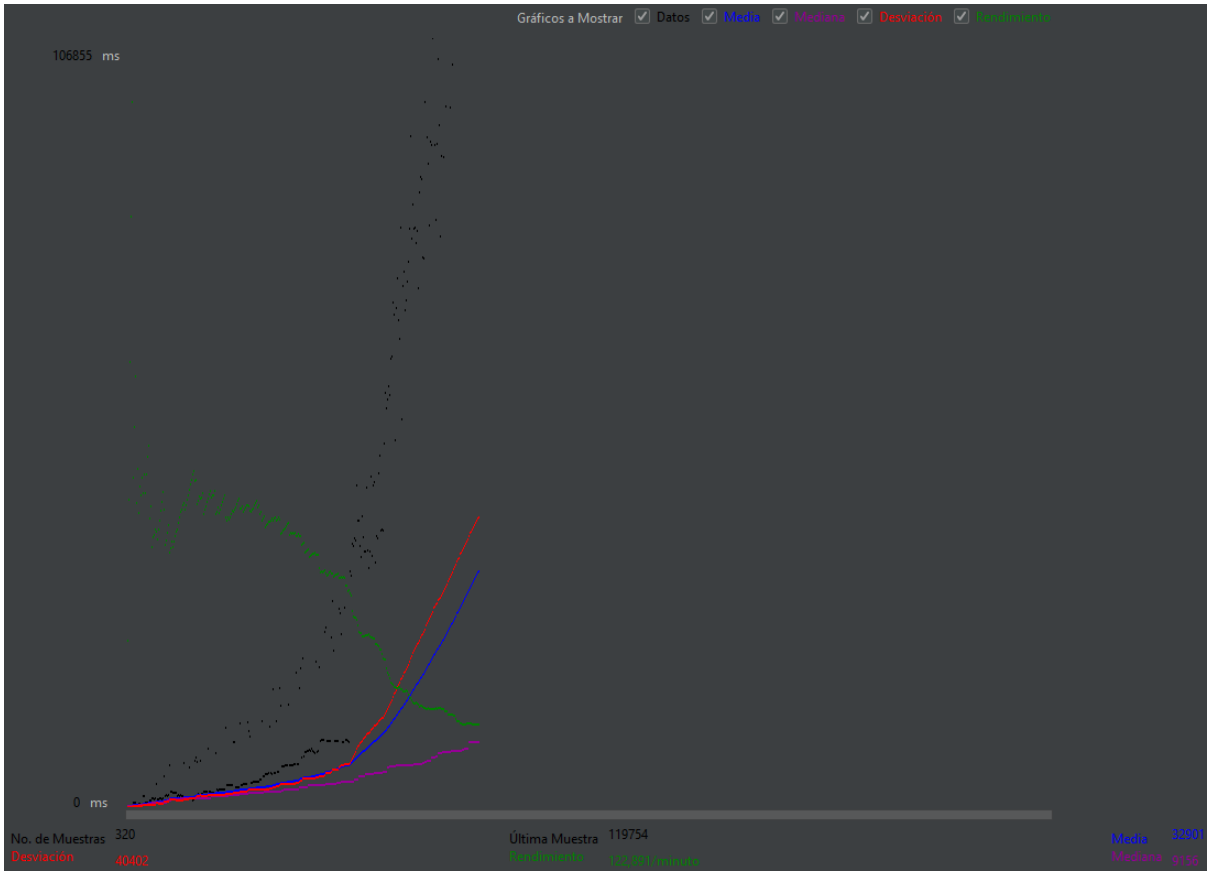


JMeter ayuda al usuario a recopilar información sobre la ejecución de sus las rutinas generadas. En la siguiente tabla generada por JMeter se evidencia 0% de error en todas las peticiones realizadas tanto al endpoint de Login como el de procesamiento de video. Sin embargo, hay un tiempo de respuesta aproximadamente de 50% más con respecto a las pruebas realizadas con una sola instancia:

Etiqueta	# Muestras	Media	Min	Máx	Desv. Estándar	% Error	Rendimiento	Kb/sec	Sent KB/sec	Media de Bytes
Login	160	3820	242	9453	2842,51	0,00%	4,2/sec	2,05	1,28	493,6
Create_task	160	61983	1669	138593	39563,06	0,00%	1,0/sec	0,21	11480,48	211,2
Total	320	32901	242	138593	40402,79	0,00%	2,0/sec	0,70	11462,20	352,4

Etiqueta	# Muestras	Media	Min	Max	Desv. Estandar	% Error	Rendimiento (s)	Kb/sec	Sent KB/sec	Medua Bytes
Login	160	3820	242	9453	2842,51	0,00%	4,2	2,05	1,28	493,6
Create_task	160	61983	1669	138593	39563,06	0,00%	1	0,21	11480,48	211,2
Total	320	32901	242	138593	40402,79	0,00%	2	0,7	11462,2	352,4

En la siguiente grafica se ve como el tiempo de ejecución aumenta (puntos negros) a lo largo de la prueba mientras que también se ve afectado el rendimiento con una tendencia a seguir disminuyendo.



Rendimiento: 122.891

En este punto, identificamos que, debido al tiempo de ejecución para la petición a la cola de mensajes, nuestra restricción es el message broker, ya que, con 8 hilos de ejecución distribuidos en 2 máquinas virtuales, no es posible que responda al alto número de solicitudes en el sistema y no sea posible encolar más rápido que con 1 sola instancia